IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: SAWANT et al	
Appl. No.: 10/710998	Art Unit: 2168
Filed: 08/16/2004	Examiner: MORRISON, JAY A
For: File System for Digital Processing Systems with Limited Resources	Atty. Docket: TI-36864

Amendment and Response Under 37 C.F.R. § 1.114

Mailbox RCE
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313 1450

Sir:

In response to the Final Office Action mailed 02/11/2008, Applicants submit the following amendments and remarks accompanying the Request for Continued Examination (RCE).

Amendments to the specification are in page 2 of this paper.

Amendments to the claims are reflected in the listing of claims which begin on page 3 of this paper.

Remarks begin at page number 13 of this paper.

It is not believed that extensions of time or fees for net addition of claims are required, beyond those which may otherwise be provided for in documents accompanying this paper. However, in the event that additional extensions of time are necessary to allow consideration of this paper, then such extensions of time are hereby petitioned under 37 C.F.R. § 1.136(a), and any fees required therefor (including fees for net addition of claims) are hereby authorized to be charged to Deposit Account No.: 20-0668.

Amendments to Specification

Please replace paragraph [0039] with the following amended paragraph:

Processor 110 may execute various instructions stored in main memory 170, secure RAM 131 and secure ROM 132 (examples of machine readable medium from which instructions can be read and executed) to enable a user to play various songs stored on SD card 150. It may also use the hardware accelerators 133 to speed up certain processing. Instructions and data to be executed in non-secure mode may be accessed and processed from main memory 170, while data and instructions to be executed in secure mode may be accessed and processed from secure mode components 130 only. The SD card controller 140 can be accessed in secure as well as non-secure mode.

Listing of Claims

Claim 1 (Currently Amended): A method of accessing data contained in a first file, wherein said first file is comprised in a plurality of files stored on a secondary storage, said secondary storage comprising a plurality of clusters, wherein a cluster is a basic unit of allocation for storing data related to a file and is identified on said secondary storage by a corresponding one of a plurality of identifiers, a file allocation table (FAT) indicating a corresponding set of clusters allocated to each of said plurality of files by associating the corresponding set of identifiers with each file whereby said FAT indicates that a first set of clusters are allocated to said first file by associating a first set of identifiers with said first file, each of said first set of identifiers identifying that a corresponding one of said first set of clusters is allocated to said first file, said first set of identifiers being contained in said plurality of identifiers, said FAT being stored on said secondary memory, said method being performed in a single device, said method comprising:

determining ~~a~~ said first set of identifiers by retrieving and examining said FAT stored in said secondary memory~~, wherein each of said set of identifiers identifies a corresponding one of said set of clusters allocated to said first file~~;

storing said first set of identifiers associated with said first file in a random access memory (RAM) indicating that said first set of clusters store data related to said file; and

retrieving at least a portion of said first file from said secondary storage based on said first set of identifiers stored in said RAM,

wherein said determining and said storing are performed when the content of said first file is to be retrieved from said secondary storage for processing.

Claim 2 (Currently Amended): The method of claim 1, ~~wherein each of said plurality of clusters is identified by a corresponding one of a plurality of identifiers,~~ said FAT storing

said first set of identifiers in the form of a linked list, wherein an order specified by said linked list indicates the sequence in which said set of clusters are used to store data contained in said first file, said method comprising:

traversing said linked list to retrieve said first set of identifiers in said order, wherein said storing stores said first set of identifiers in said RAM.

Claim 3 (Currently Amended): The method of claim 2, wherein said first set of identifiers are stored according to a technique which permits each of said first set of identifiers to be retrieved with fewer instructions than the number of instructions required to access the same identifier from said FAT in said secondary storage.

Claim 4 (Currently Amended): The method of claim 2, wherein said first set of identifiers are stored in the form of an array in said RAM which permits each identifier to be retrieved by a single access.

Claim 5 (Currently Amended): The method of claim 4, further comprising:

receiving a start offset of data to be accessed;

computing a cluster index by dividing said start offset by a number of bytes in each of said plurality of clusters; and

accessing said array using said cluster index to determine a specific one of said first set of identifiers, wherein said data to be accessed is present in a cluster identified by said specific one of said first set of identifiers.

Claim 6 (Currently Amended): The method of claim 5, wherein data stored in said first file represents a song and wherein said single device is designed to play said song based on the stored data, said method further comprising:

receiving a request for rewind operation requiring access to data in a previous cluster when playing said song;

accessing the data in said previous cluster using said first set of identifiers stored in said RAM .

Claim 7 (Currently Amended): A method of implementing an application in a system containing a small memory in the form of a random access memory (RAM), wherein said system supports a file system on a secondary storage, wherein said secondary storage comprises a plurality of clusters each identified by a corresponding identifier in said secondary storage, wherein said file system comprises a plurality of files and each of said plurality of files is stored in a corresponding one of a plurality of sets of clusters, a file allocation table (FAT) stored on said secondary storage indicating a corresponding set of clusters allocated to each of said plurality of files by associating corresponding set of identifiers with each file, said plurality of sets of clusters being contained in said plurality of clusters, said method comprising:

providing a first module which is designed to determine a plurality of identifiers corresponding to a specified file by examining said FAT and store said plurality of identifiers in ~~a random access memory (RAM)~~ said RAM according to a convention, wherein said plurality of identifiers specify a set of clusters corresponding to said specified file, said set of clusters being contained in said plurality of sets of clusters;

providing a second module which is to perform an operation on a file of interest, wherein said second module is designed to determine a desired cluster by using said plurality of identifiers stored in said RAM according to said convention;

executing said first module when the content of said specified file is to be retrieved from said secondary storage for processing such that only a portion of said FAT including data indicating that said plurality of identifiers identify the clusters storing data related to said specified file is stored in said small memory,

wherein said first module is executed while specifying said file of interest as said specified file such that said plurality of identifiers corresponding to said file of interest are stored in said RAM according to said convention; and

executing said second module after executing said first module,

wherein both of said first module and said second module are executed using at least some of the same locations of said small memory.

Claim 8 (Original): The method of claim 7, wherein said second module is overlaid on the same memory space on which said first module is loaded during execution.

Claim 9 (Previously Presented): The method of claim 8, wherein said convention comprises storing said plurality of identifiers at a pre-specified portion of said RAM.

Claim 10 (Original): The method of claim 9, wherein each of said plurality of files stores data representing a corresponding song.

Claim 11 (Currently Amended): A machine readable medium carrying one or more sequences of instructions for causing a single digital processing system to access data

contained in a first file, wherein said first file is comprised in a plurality of files stored on a secondary storage, said secondary storage comprising a plurality of clusters each identified on said secondary storage by a corresponding one of a plurality of identifiers, a file allocation table (FAT) indicating a corresponding set of clusters allocated to each of said plurality of files by associating the corresponding set of identifiers with each file whereby said FAT indicates that a first set of clusters are allocated to said first file by associating a first set of identifiers with said first file, said first set of identifiers being contained in said plurality of identifiers, said FAT being stored on said secondary memory, wherein execution of said one or more sequences of instructions by one or more processors contained in said digital processing system causes said one or more processors to perform the actions of:

determining ~~a~~ said first set of identifiers by retrieving and examining said FAT stored in said secondary memory~~, wherein each of said set of identifiers identifies a corresponding one of said set of clusters allocated to said first file~~;

storing said first set of identifiers associated with said first file in a random access memory (RAM) indicating that said first set of clusters store data related to said file; and

retrieving at least a portion of said first file from said secondary storage based on said first set of identifiers stored in said RAM,

wherein said determining and said storing are performed when the content of said first file is to be retrieved from said secondary storage for processing.

Claim 12 (Currently Amended): The machine readable medium of claim 11, wherein each of said plurality of clusters is identified by a corresponding one of a plurality of identifiers, said FAT storing said first set of identifiers in the form of a linked list, wherein an order specified by said linked list indicates the sequence in which said set of clusters are used to store data contained in said first file, further comprising:

traversing said linked list to retrieve said first set of identifiers in said order, wherein said storing stores said first set of identifiers in said RAM.

Claim 13 (Currently Amended): The machine readable medium of claim 12, wherein said first set of identifiers are stored according to a technique which permits each of said first set of identifiers to be retrieved with fewer instructions than the number of instructions required to access the same identifier from said FAT in said secondary storage.

Claim 14 (Currently Amended): The machine readable medium of claim 12, wherein said first set of identifiers are stored in the form of an array in said RAM which permits each identifier to be retrieved by a single access

Claim 15 (Canceled)

Claim 16 (Currently Amended): The machine readable medium of claim 14, further comprising:

receiving a start offset of data to be accessed;

computing a cluster index by dividing said start offset by a number of bytes in each of said plurality of clusters; and

accessing said array using said cluster index to determine a specific one of said first set of identifiers, wherein said data to be accessed is present in a cluster identified by said specific one of said first set of identifiers.

Claim 17 (Currently Amended): A machine readable medium carrying one or more sequences of instructions for causing a digital processing system to implement an application

using a small memory space in a random access memory (RAM), wherein said digital processing system supports a file system on a secondary storage, wherein said secondary storage comprises a plurality of clusters each identified by a corresponding identifier in said secondary storage, wherein said file system comprises a plurality of files and each of said plurality of files is stored in a corresponding one of a plurality of sets of clusters, said plurality of sets of clusters being contained in said plurality of clusters, a file allocation table (FAT) stored in said secondary storage indicating a corresponding set of clusters allocated to each of said plurality of files, wherein execution of said one or more sequences of instructions by one or more processors contained in said digital processing system causes said one or more processors to perform the actions of:

providing a first module which is designed to determine a plurality of identifiers corresponding to a specified file by examining said FAT and store said plurality of identifiers in ~~a random access memory (RAM)~~ said RAM according to a convention, wherein said plurality of identifiers specify a set of clusters corresponding to said specified file, said set of clusters being contained in said plurality of sets of clusters;

providing a second module which is to perform an operation on a file of interest, wherein said second module is designed to determine a desired cluster by using said plurality of identifiers stored in said RAM according to said convention;

executing said first module when the content of said specified file is to be retrieved from said secondary storage for processing such that only a portion of said FAT including data indicating that said plurality of identifiers identify the clusters storing data related to said specified file is stored in said small memory,

wherein said first module is executed while specifying said file of interest as said specified file such that a plurality of identifiers corresponding to said file of interest are stored in said RAM according to said convention; and

executing said second module after executing said first module,

wherein both of said first module and said second module are executed using at least some of the same locations of said small memory.

Claim 18 (Original):The machine readable medium of claim 17, wherein said second module is overlaid on the same memory space on which said first module is loaded during execution.

Claim 19 (Previously Presented):The machine readable medium of claim 18, wherein said convention comprises storing said plurality of identifiers at a pre-specified portion of said RAM.

Claim 20 (Currently Amended): An apparatus accessing data contained in a first file, wherein said first file is comprised in a plurality of files stored on a secondary storage, said secondary storage comprising a plurality of clusters, wherein a cluster is a basic unit of allocation for storing data related to a file and is identified on said secondary storage by a corresponding one of a plurality of identifiers, a file allocation table (FAT) indicating a corresponding set of clusters allocated to each of said plurality of files by associating the corresponding set of identifiers with each file whereby said FAT indicates that a first set of clusters are allocated to said first file by associating a first set of identifiers with said first file, each of said first set of identifiers identifying that a corresponding one of said first set of clusters is allocated to said first file, said first set of identifiers being contained in said plurality of identifiers, said FAT being stored on said secondary memory, said apparatus comprising:

means for determining ~~a~~ said first set of identifiers by retrieving and examining said FAT stored in said secondary memory~~, wherein each of said set of identifiers identifies a corresponding one of said set of clusters allocated to said first file~~;

means for storing said first set of identifiers associated with said first file in a random access memory (RAM) indicating that said first set of clusters store data related to said file; and

means for retrieving at least a portion of said first file from said secondary storage based on said first set of identifiers stored in said RAM,

wherein said determining and said storing are performed when the content of said first file is to be retrieved from said secondary storage for processing.

Claim 21 (Currently Amended): The apparatus of claim 20, ~~wherein each of said plurality of clusters is identified by a corresponding one of a plurality of identifiers,~~ said FAT storing said first set of identifiers in the form of a linked list, wherein an order specified by said linked list indicates the sequence in which said set of clusters are used to store data contained in said first file, said apparatus further comprising:

means for traversing said linked list to retrieve said first set of identifiers in said order, wherein said storing stores said first set of identifiers in said RAM.

Claim 22 (Currently Amended): The apparatus of claim 21, wherein said first set of identifiers are stored according to a technique which permits each of said first set of identifiers to be retrieved with fewer instructions than the number of instructions required to access the same identifier from said FAT in said secondary storage.

Claim 23 (Currently Amended): The apparatus of claim 21,wherein said first set of identifiers are stored in the form of an array in said RAM which permits each identifier to be retrieved by a single access.

Claim 24 (Canceled)

Claim 25 (Currently Amended): The apparatus of claim 24, further comprising:

means for receiving a start offset of data to be accessed;

means for computing a cluster index by dividing said start offset by a number of bytes in each of said plurality of clusters; and

means for accessing said array using said cluster index to determine a specific one of said first set of identifiers, wherein said data to be accessed is present in a cluster identified by said specific one of said first set of identifiers.

Claims 26 - 28 (Canceled)

REMARKS

Claims 1-26 were examined in the outstanding final office action mailed on 02/11/2008 (hereafter "First Final Office Action"). The specification was objected to and all the claims were rejected.

By virtue of this paper, the specification and claims 1-7, 11-14, 16-17, 20-23 and 25 are sought to be amended and claims 15 and 24 are sought to be canceled. The amendments and cancellations are believed not to introduce new matter and their entry is respectfully requested. Claims 1-14, 16-23 and 25 are thus respectfully presented for reconsideration.

Specification

In page 2 paragraph 2 of the First Final Office Action, the specification was objected to as failing to provide proper antecedent basis for the claimed subject matter. In particular, it was alleged that the term "machine readable medium" as claimed in the preamble of claims 11-19 is not defined in the specification.

Without acquiescing to any of the Examiner's allegations, it is respectfully noted that the specification is sought to be amended to provide the requested antecedent basis. The amendments are believed not to introduce matter and their entry is respectfully requested. Withdrawal of the objection with respect to the specification is respectfully requested.

Claim Rejections - 35 U.S.C. § 101

In paragraphs 3 and 4 of the Outstanding Office Action, the Examiner had indicated that claims 1-10 constitute patentable subject matter under 35 U.S.C. § 101. The Examiner is thanked for the same.

Claim Rejections - 35 U.S.C. § 103

Claims 1-4,11-14, 20-23 were rejected under 35 U.S.C. 103(a) as being unpatentable over Suzuki (Patent Number 6,604,170) in view of Nelson *et al* ('Nelson' hereinafter) ('Caching in the Sprite Network File System", ACM Transactions on Computer-Systems, Vol. 6, No. 1, February 1988, pages 134-154).

Without acquiescing to any of the Examiner's contentions it is respectfully submitted that the submitted are allowable over the art of record.

For example, currently amended claim 1 recites that the claimed cluster identifiers respectively **identify corresponding clusters on the secondary memory**, the FAT (containing the **identifiers associated with individual files** to indicate the clusters storing the data of the respective files) is stored on the secondary memory and the **RAM stores the same information (as in the FAT) as pertinent to a first file** when the file is sought to be accessed. Furthermore, all the elements of currently amended claim 1 are performed in a single device.

The Examiner relies on Nelson to teach the claimed storing of the identifiers in a random access memory (RAM). It is respectfully noted that Nelson, either alone or in combination with other references of record, does not teach or reasonably suggest at least the above noted features of currently amended claim 1.

As a basis for such an assertion, Applicants first note that the disclosure of Nelson relied upon by the Examiner relates to a situation in which data is retrieved by a client system via a server. In sharp contrast, currently amended claim 1 relates to performance of all the actions in a single device and also the actions are recited to be performed **when the content of the file is to be retrieved**.

Furthermore, in accordance with the invention of currently amended claim 1, the cluster identifiers identify the specific clusters (basic units of allocation) in the secondary storage and also the FAT is stored in the secondary storage. The same information as pertinent to an individual file is retrieved and stored in the RAM.

At least the portions of Nelson do not clarify various details such as the specific units of allocation in secondary storage, for the Applicant to clearly address whether the rejection under 35 U.S.C. § 103 would be proper at least as against currently amended claim 1.

In particular, the Examiner's attention is drawn to the disclosure of Nelson stating, "... We used **virtual addresses instead of physical disk addresses** so that clients could create

new blocks in their caches without first contacting a server to find out their physical locations..." (Page 138, lines 2-4 of Nelson, **Emphasis Added**). This teaching, at least when read alone, appears to impel a skilled practitioner not to combine Suzuki and Nelson as in currently amended claim 1.

The burden of coming forward with sufficient information rests with the Patent Office under the applicable laws/practice and it is accordingly submitted that the portions of Nelson relied upon by the Examiner would not establish a proper basis of rejection under 35 U.S.C. § 103.

Currently amended claim 1 is accordingly believed to be allowable over the art of record. Claims 2-6 depend from currently amended claim 1, and are allowable at least for reasons noted above with respect to claim 1.

Claim 4 is independently allowable at least over the portions relied upon by the Examiner in reciting that the identifiers of the clusters storing the data of the first file are stored **in the RAM in the form of an array**, permitting each identifier to be accessed in a **single access**.

The Examiner relies on Col. 6 lines 60-65 of Suzuki in rejecting claim 4 (see Page 6 lines 1-2 of the First Final Office Action). It is respectfully noted that this portion of Suzuki teaches neither an array not storing the array in a RAM at least based on the below highlighted portion:

> FIG. 3 shows the ***structure of the directory entry used to record FAT information of each file*** in accordance with the cluster allocation order. This directory entry will be referred to as an EXT-FAT in distinction from the conventional FAT 1302 (FIG. 13). In the EXT-FAT, items corresponding to FAT items will be referred to as EXT-FAT items.
>
> The start byte ((00h)-th byte) of the **EXT-FAT records an index in which "1" is** set in bit 7, so as to be distinguished from the two different directory entries shown in FIGS. 4 and 5 to be described later, and to maintain compatibility with the conventional FAT file system. Bytes 0Bh to 0Dh, and 1Ah and 1Bh are used as reserved areas to maintain compatibility with the conventional directory entry structure. The remaining slots record cluster addresses allocated to a given file in the allocation order from a lower cluster address in correspondence with the number of slots. In this embodiment, a cluster address

> is expressed by 16 bits (2 bytes). Hence, the EXT-FAT of this embodiment can record FAT information for 13 clusters. (Col. 6 lines 60-65 of Suzuki, ***Emphasis Added***)

From the above, it is concluded that the directory entry of Fig. 3 of Suzuki is written on a secondary storage (not a RAM) and thus currently amended claim 4 is independently allowable over the portions relied upon by the Examiner.

Currently amended claim 6 is also independently allowable in applying the features to a device which plays songs (likely to have limited memory). In particular, the features are specifically applied to have quick access to data (song portion) in a prior cluster when a rewind operation is to be performed.

Currently amended independent claim 7 is also allowable over the art of record in reciting a combination of features including that the identifiers identify the cluster **on the secondary storage**, and not all the entire FAT is stored in the RAM, but only a portion of the FAT including the specific identifiers identifying clusters of a specific file are stored when the specific file is sought to be accessed.

By storing only a portion of the FAT (including only the required identifiers), the **small** memory may be efficiently utilized.

The portions of Suzuki and Nelson, relied upon by the Examiner, do not teach or reasonably suggest such a combination of features.

For example, equating the storing of "virtual addresses" (see lines 2-5 Page 135 of Nelson) of Nelson with the claimed identifiers would be factually erroneous since the virtual addresses do not identify the clusters **in the secondary storage**.

Similarly, equating the cacheing of 'file maps' (see lines 7-10 page 138 of Nelson) of Nelson with the claimed storing would also be factually erroneous since there is no teaching or suggestion that the file map is cached when a specific file is to be accessed or that only a portion of the file map is cached, even assuming arguendo that the file map of Nelson is akin to the claimed FAT.

At least for some of such reasons, currently amended claim 7 is allowable over the art of record. Claims 8-10 depend from claim 7 and are allowable at least for the reasons noted above with respect to claim 7.

The remaining currently amended independent claims 11 and 20 are allowable over the art of record at least for some of the reasons noted above. The dependent claims are allowable at least as depending from the corresponding allowable base claims.

Conclusion

Accordingly all the objections and rejections of record are believed to be overcome. Continuation of examination is respectfully requested. The Examiner is invited to telephone the undersigned representative at 707.356.4172 if it is believed that an interview might be useful for any reason.

Respectfully submitted,

/Narendra Reddy Thappeta/
Signature
Printed Name: Narendra Reddy Thappeta
Attorney for Applicant
Registration Number: 41,416

Date: May 26, 2008